1. Application Architecture

```
┌─────────────────┐      ┌─────────────────┐      ┌─────────────────┐
│ JavaServer Faces│      │   Enterprise    │      │ Java Persistent │
│                 │─────▶│   JavaBeans     │─────▶│      API        │
│ (Presentation   │      │                 │      │                 │
│    Layer)       │      │ (Business Layer)│      │                 │
└─────────────────┘      └─────────────────┘      └─────────────────┘
                                                           │
                                                           ▼
                                                  ┌─────────────────┐
                                                  │     JavaDB      │
                                                  │                 │
                                                  └─────────────────┘
```
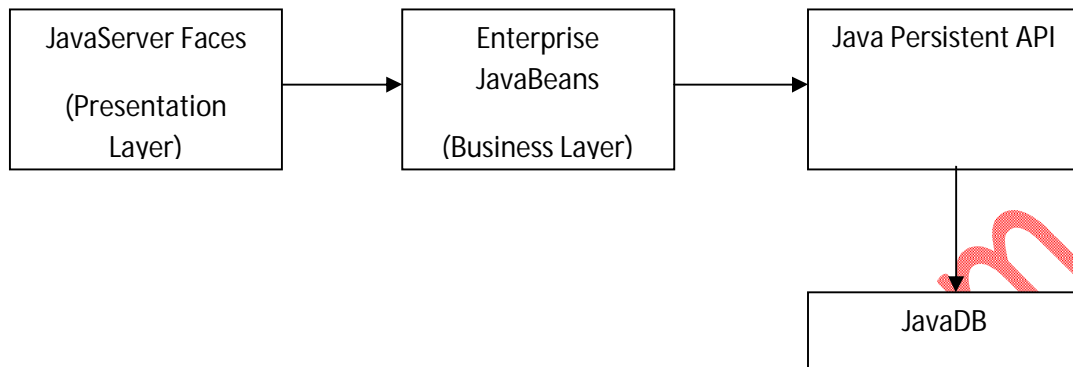
2. Interaction Between Tiers

The application has been organized into four tiers Presentation, Business, Data manipulation and data, each layer plays a specific role and is segregated from another.

The JavaServer faces are used for the presentation layer. Its purpose is to get data or send data to EJB (business layer). The JavaServer faces provide interaction between user and application. When user clicks on list property a request is sent to EJB which in turn provides data using JPA.

Enterprise JavaBeans are the core of the application. This layer receives request from the JSF, process it and then passes it on to JPA in turn JPA run queries and send data to to EJB which again forward it to the presentation layer.

JPA is the data access layer which provides the access to the JavaDB and serves the requests of the EJB

JavaDB stores the structured data, which JPA uses to fetch different forms of data.

3. Page Navigation

There are several pages in the application. To link them together managed beans are used. Managed beans are used to go from one page to another. Managed beans uses backed beans to hold the data and properties of the components.

After user enters a property in add property page and clicks "Add" button at this point the Managed Bean gets activated and sends the control to the next page which is

propertylists.xhtml, a "Property added successfully" message is shown to the user and a lists of properties are displayed to the user.

Then there are options to go back to main page on every sub pages which users enters from the main page.

4. Tests

Test	Expected results	Actual Results
To check if Property is added to the database	Property gets persistent	Property gets persistent
To test if Property List gets displayed	Lists gets displayed	Lists get displayed
To test if Agency is added to the database	Agency is added	Agency is added
To test if Agency Lists us displayed	Lists displayed	List displayed
To check if Property Manager is added to the database	Property Manager is added	Property Manager is added
To test if list gets displayed	Lists gets displayed	Lists get displayed